21.1 Free-Running Ring Frequency Synthesizer

Daniel J. Allen, Adam L. Carley

TimeLab, Andover, MA

PLLs have long been the mainstay of clock synthesizer designs. Architectural innovations that responded to the need for better frequency-setting resolution, transient response, and noise immunity have increased the complexity and therefore the cost and power consumption of synthesizer implementations [1]. A proposed approach to clock synthesis is described that overcomes the limitations of PLLs while adding numerous features. The timing processor unit (TPU™) employs this approach to integrate 6 independent high resolution low-jitter synthesizers with spread-spectrum modulation onto a single die using a low-cost 0.18µm CMOS process.

In contrast to a PLL, which controls the loop, the TPU measures a free-running ring oscillator (FRO) period against a reference clock (RefClk). For each output clock cycle, the TPU uses the measured period to determine which FRO outputs (Taps) will be selected to generate the required clock edges. Each selected Tap passes through a digital delay line, or vernier, which provides precision between Taps. The TPU then uses the delayed Tap to produce the desired output clock.

As illustrated in Figures 21.1.1 and 21.1.2, the TPU uses an FRO of cross-coupled differential delay elements. Signals Tap0 through Tap15 emerge from the junctions between the successive stages of the FRO. The Lap Counters tally the cycles (Laps) of Tap0 and Tap8. The FRO and Lap Counters together constitute a short-term stable high precision oscillator (timing changes due to PVT variations occur slowly).

The Capture block measures the FRO period against the external RefClk period. It captures and processes the Laps and Taps on successive rising RefClk edges to eliminate metastability effects; the result is then binary encoded. Metastability in the capture of the Taps is eliminated by comparing the captured levels with expected patterns. Captured Tap0 is used to select the known valid Lap capture, removing metastability effects in the capture of the Laps. The difference between subsequent binary-encoded values determines the count of Taps per RefClk period (TapsPerRefClk). The TapsPerRefClk is updated on every RefClk and processed through a series of programmable filters to generate an average TapsPerRefClk (AvgTapsPerRefClk) with sub-Tap precision to 1/1024th of a Tap.

The Processor block calculates the Lap, Tap, and vernier delay for each output clock edge. Each output clock frequency is specified as a ratio of its period relative to the RefClk period. This ratio is then multiplied by the AvgTapsPerRefClk to determine the Taps per output clock (TapsPerOutClk), and the Processor block accumulates the TapsPerOutClk at the output rate to drive the Channel block.

The Channel block decodes the Lap, Tap, and vernier data from the Processor block to generate the output clock. The Tap data selects the Tap edge that drives the vernier, and the vernier data sets the sub-tap delay of the edge through the vernier. The vernier then sends the edge to the clock of the JK flip-flop. Based on timing analysis of the layout, the Taps are divided into 2 groups where the Tap transition and Lap comparison can reliably meet the setup and hold times of the JK flip-flop. The flip-flop is enabled to toggle when the Lap data matches the current Lap Counter, thus creating an output clock edge.

Figure 21.1.3 illustrates that each Tap signal is delayed approximately 100ps from its progenitor, producing an oscillator frequency of about 625MHz. Normal PVT variations mean that the tap delays may vary between 71ps and 162ps. Selected Taps are delayed by the vernier and JK propagation delay to produce the output clock. The vernier allows better timing precision for the synthesized clock edges than the tap spacing. The vernier has a 6b input to select equally spaced delays of 1/64th of a tap delay. By design, the vernier timing tracks the tap timing within ±3% over PVT. The vernier delay setting is updated for every output clock edge based on the value in the accumulator, which carries extra bits of precision in its calculations to maintain high precision of the synthesized clock periods.

An example of the TPU process can be illustrated using an external reference clock of 25MHz that has a RefClk period of 40ns. A value of 1/8 is programmed into the period register to create a 200MHz clock. Using a 19b fractional binary depth provides uniform 76fs output clock period steps; therefore, over the 12.5 to 400MHz frequency range, the LSB step-size range is 47Hz to 12.2kHz.

Each output channel can generate a clock of up to 66MHz, and multiple channels can be combined to produce higher frequency clocks; test chips have demonstrated over 2GHz operation. The TPU has 6 independent synthesizers driven by a single FRO, including a 6 channel synthesizer that operates up to 400MHz.

The digital design of the TPU enables precise and predictable modulation of the output clock frequency without over or undershoot. The spread-spectrum modulation profile, shown in Fig. 21.1.4, is accomplished with a simple state machine that increments and decrements the synthesizer period by a programmed amount at a programmed rate.

In addition to modulation capabilities, Fig. 21.1.4 also illustrates the TPU synthesizer slewing of clocks between different frequencies, controlled by a slew rate register and an input signal named Auto-F™. Figure 21.1.5 illustrates simultaneous clock frequency slewing of three TPU outputs. The TPU slew rate uses fine-precision steps and can be programmed for the particular system. For example, since system components often include PLLs designed to track 33kHz spread-spectrum modulation, the slew rate can be programmed slower than 33kHz to prevent downstream PLLs from losing lock, providing consistent and reliable system operation.

The jitter characteristics of the TPU outputs differ from PLLs in that the TPU does not accumulate error from cycle to cycle. The frequency of the significant jitter is at or above the 25MHz reference clock frequency. When the TPU outputs are used as a reference clock to a PLL, the loop filter of the PLL attenuates the TPU jitter, resulting in improved overall performance. The jitter spectrum of the TPU is illustrated in Fig. 21.1.6, both before and after MATLAB processing with the PCI SIG jitter mask. The inverse FFT of the processed jitter spectrum confirms 76ps of data eye closure attributed to clock jitter, well within the PCI Express specification of 86ps [1].

Previous digital approaches to synthesis have been limited by frequency range, precision and jitter. With edge addressability better than 3ps and fine frequency-setting precision, all PC system reference clocks can be generated from a single FRO. The TPU design eliminates the problem of integrating many PLLs onto a single die and enables system designers to dynamically change clocks without interrupting CPU execution, which reduces power consumption in portable and desktop systems while maximizing overclocking performance in high-end platforms.

References:

[1] G. Yan, et al., "A Self-Biased PLL With Current-Mode Filter for Clock Generation," *ISSCC Dig. Tech. Papers,* pp. 420-421, Feb., 2005.

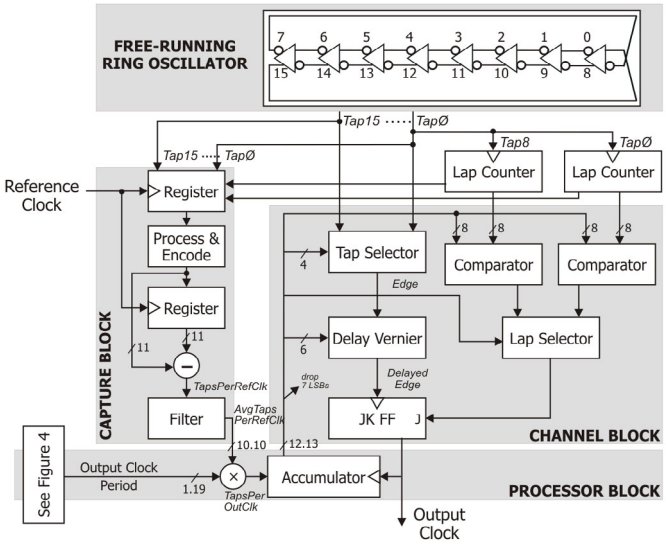

Figure 21.1.1: TPU functional blocks.

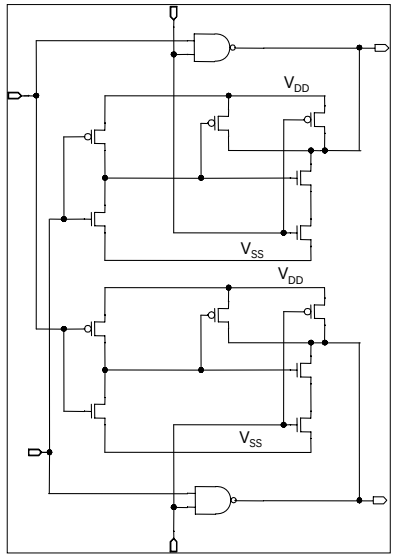

Figure 21.1.2: Cross-coupled delay element.

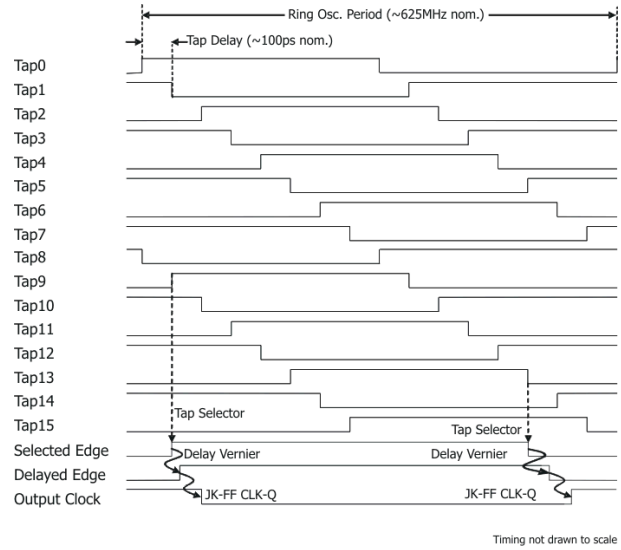

Figure 21.1.3: Tap waveforms and selection.

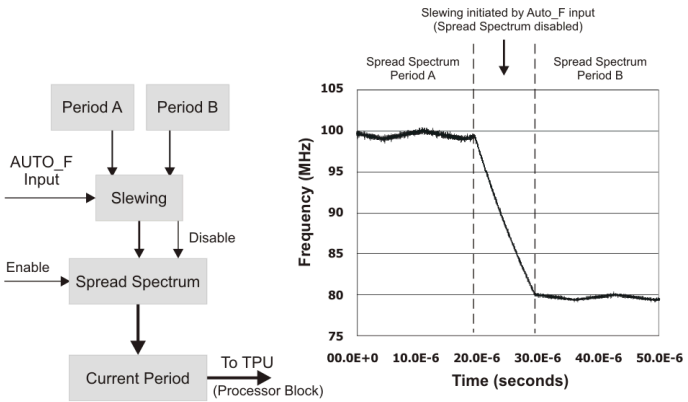

Figure 21.1.4: Frequency slewing and spread spectrum.

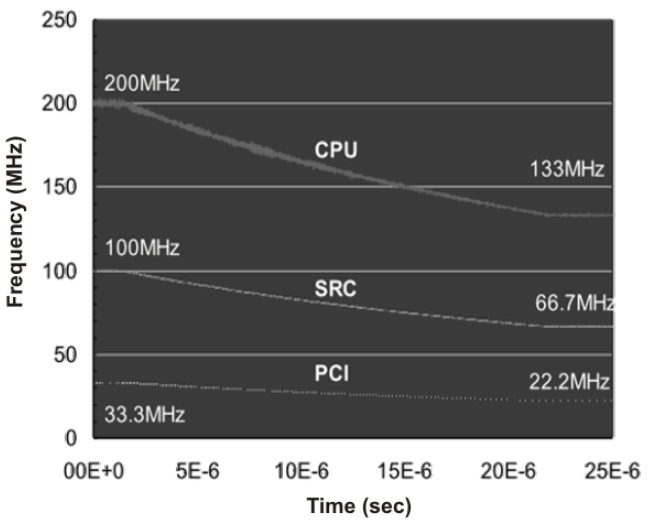

Figure 21.1.5: Simultaneous slewing of multiple clocks.

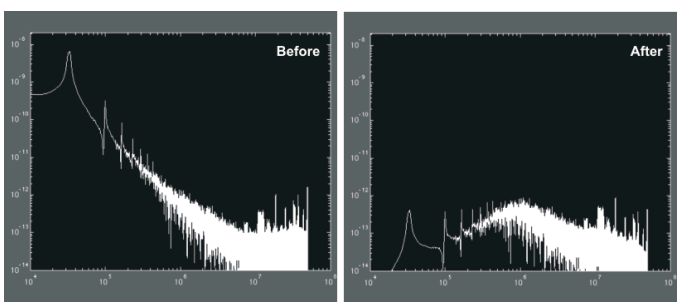

Figure 21.1.6: Jitter analysis.

Continued on Page 660

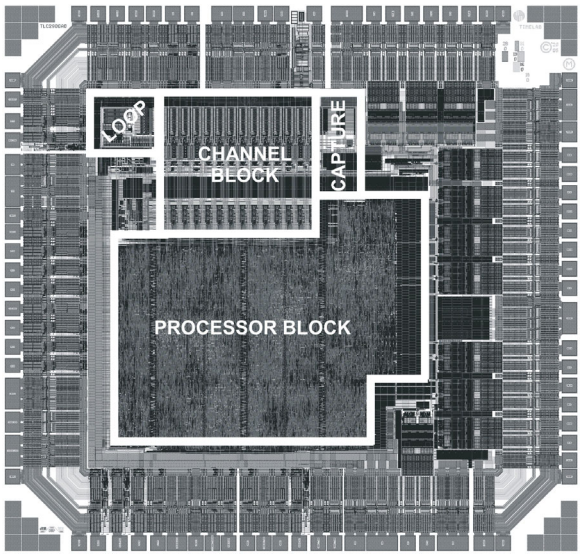

Figure 21.1.7: Die micrograph.

